hypertext transfer protocol; (2) the access file invokes at least one service module, which passes the one or more control parameters and the service deliverable to the communication module in accordance with a hypertext transfer protocol; and (3) the communication module transmits the service deliverable received from the at least one service module to the destination node in any format selected from a voice format, an internet format, an e-mail format, and a wireless format.

The Examiner has asserted that:

> Giese teaches the invention substantially as claimed including a system for providing remote electronic services [col. 1, lines 15-20], comprising an agent [10, Fig. 9], an access file [service triggers, Fig. 11; col. 5, lines 23-27 & 49-54], and a communication module [14, Fig. 9] ...

That is, the Examiner has identified Giese's contact agent 10 as corresponding to the agent recited in claim1, the transport agent 14 as corresponding to the communication module recited in claim 1, and the "service triggers" issued by the contact agent 10 as corresponding to the access file that invokes at least one service module that produces a service deliverable. The Examiner also has identified Giese's exchange agent 12 as corresponding to the at least one service module recited in claim 1.

The Examiner has acknowledged that Giese fails to teach or suggest that communications between the contact agent 10, the exchange agent 12, and the transport agent 14 are in accordance with a hypertext transport protocol. Nevertheless, the Examiner has concluded that:

> ... it would have been obvious to a person of ordinary skill in the art at the time the invention was made to utilize HTTP format in Giese's system because http is a well-known protocol in the art for generating service query over network. One of ordinary skill in the art would have been motivated to modify Giese's system with http format based on specific design requirements.

1. The Examiner has failed to establish a proper *prima facie* case of obviousness

For the purpose of the following discussion, the examiner is reminded that:

> To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the

> knowledge generally available to one of ordinary skill in the art, to modify the references or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not on applicants' disclosure.

MPEP § 706.02(j). Furthermore, as pointed out by the Patent Office Board of Appeals and Interferences:

> The examiner should be aware that "deeming" does not discharge him from the burden of providing the requisite factual basis and establishing the requisite motivation to support a conclusion of obviousness.

Ex parte Stern, 13 USPQ2d 1379 (BPAI 1989).

With his rejection, the Examiner has failed to provide the requisite factual basis and failed to establish the requisite motivation to support his deemed conclusion that the features recited in independent claim 1 would have been obvious to one of ordinary skill in the art at the time of the invention. The Examiner merely asserts without any basis that "one of ordinary skill in the art would have been motivated to modify Giese's system with http format based on specific design requirements." The Examiner, however, does not even hint at the nature of those "specific design requirements" that would have motivated one of ordinary skill in the art to modify the agents disclosed in Giese. Moreover, that Examiner has failed to point to any "specific design requirements" taught or suggested in Giese that would have led one of ordinary skill in the art to modify Giese's agents 10-14 in the way the Examiner has proposed. For these reasons, the Examiner has failed to establish a proper *prima facie* case of obviousness, as required under MPEP § 706.02(j).

The Examiner is requested to cite prior art in support of his assertions. Alternatively, if the Examiner is aware of facts within his personal knowledge that provide the requisite factual basis and establishes the requisite motivation to support his deemed conclusion that the features recited in independent claim 1 would have been obvious, the Examiner is requested to provide an affidavit in accordance with 37 CFR § 1.104(d)(2). Otherwise, the Examiner's rejection of independent claim 1 should be withdrawn.

2. In any event, the invention of claim1 is not obvious over Giese

a. Overview

Giese's various agents 10, 12, 14 communicate via communication and network primitives, not networking protocols. In particular:

> User applications in the content application layer 4, the Contact Agent 10 and adjacent networks 48 interact with the Exchange Agent 12 using communication primitives. The Exchange Agent 12 and adjacent networks interact with the Transport Agent 14 using network primitives. (Col. 16, lines 7-11)
>
> ...
>
> Primitives must be simple and easy to use and, for the most part, concise and limited in number. Their format is not unlike software language instructions in that they consist of an `instruction code` part followed by a string of one or more arguments. ... (Col. 16, lines 47-51)

In general, communications between agents on a single machine are handled by function calls to the operating system, whereas communications between agents on different machines are handled by networking protocols. In Giese's system, the contact agent 10, the exchange agent 12, and the transport agent 14 are executed on a single machine upon initiation of a service instance. Therefore, the communication from the contact agent 10 to the exchange agent 12 and the communication between the exchange agent 12 and the transport agent 14 are handled by function calls to the operating system of the machine. These communications do not involve the use of any type of networking protocol because these communications are not over a network.

HTTP is a networking protocol used for communications between content applications in the application layer. In particular, HTTP is an application layer protocol that requires a connection-oriented transport layer protocol (i.e., TCP) to guarantee the delivery of packets. As explained in the preceding paragraph, there are no network connections between the contact agent 10 and the exchange agent 12, nor are there any network connections between the exchange agent 12 and the transport agent 14. Therefore, no one with any skill in the field of network communications would configure communications from the contact agent 10 to the exchange agent 12 in accordance with the HTTP protocol, nor would one with any skill in the field of network communications configure communications between the exchange agent 12 and the transport agent 14 in accordance with the HTTP protocol. Indeed, even assuming for the purpose of argument that it would be possible to configure such communications in accordance with the HTTP protocol, at best such a configuration would be a terribly inefficient way to configure communications between these agents.

b. Giese's contact agent 10

Giese does not teach or suggest that the contact agent 10 is operable to transmit a request-for-service call to an access file in accordance with a hypertext transfer protocol for each of the received request-for-service calls, as recited in claim 1. Giese teaches that the contact agent 10 transmits originating and receiving party profiles to the exchange agent 12 (see col. 12, lines 20-22). However, the only disclosure in Giese regarding the format of communications between the contact agent 10 and the exchange agent 12 is as follows:

> User applications in the content application layer 4, the Contact Agent 10 and adjacent networks 48 interact with the Exchange Agent 12 using communication primitives. ... (Col. 16, lines 7-9)
>
> Communication primitives are abstractions that enable users, user agents or adjoining networks to interact with the ECS to initiate, maintain and terminate communication services for a wide variety of content. The communication primitives consist of basic instructions sent to the Exchange Agent 12, and subsequent responses returned by the Exchange Agent 12 to enable dialog with ECS functions. The communication primitives enact fundamental communication behaviors from which more complex services can be derived. (Col. 16, lines 18-27)

There is no hint in this disclosure that would have led one of ordinary skill in the art at the time of the invention to structure communications from the contact agent 10 to the exchange agent 12 in accordance with a hypertext transfer protocol. Indeed, as explained in the Overview above, the contact agent 10 and the exchange agent 12 are on a single machine and therefore do not communicate over a network connection. For this reason, there is no reason whatsoever for one of ordinary skill in the art to modify the communications between the contact agent 10 and the exchange agent 12 to be in accordance with a networking protocol, much less to modify such communications in accordance with an application layer networking protocol such as HTTP.

c. Giese's exchange agent

The Examiner also has asserted that the service module invoked by the service triggers issued by the contact agent 12 corresponds to the exchange agent 12. Contrary to the Examiner's assertion, however, Giese does not teach or suggest that the exchange agent 12 is operable to pass one or more control parameters and a service deliverable to a communication module in accordance with a hypertext transfer protocol, as recited in claim 1.

Giese teaches that the exchange agent 12 interacts with the transport agent 14 (which the Examiner has asserted corresponds to the communication module recited in claim 1) using network primitives (see col. 16, lines 9-11). Giese describes the nature and function of such network primitives at col. 16, lines 28-62. There is nothing in this disclosure, however, that would have led one of ordinary skill in the art at the time of the invention to structure communications between the exchange agent 12 and the transport agent 14 in accordance with a hypertext transfer protocol. Indeed, as explained in the Overview above, the exchange agent 12 and the transport agent 14 are on a single machine and therefore do not communicate over a network connection. For this reason, there is no reason whatsoever for one of ordinary skill in the art to modify the communications between the exchange agent 12 and the transport agent 14 to be in accordance with a networking protocol, much less to modify such communications in accordance with an application layer networking protocol such as HTTP.

In addition, Giese's exchange agent 12 does not produce a service deliverable in accordance with a request-for-service call, as recited in claim 1. In Giese's approach, the service deliverable is produced by a content application operating in the application layer 4, not the exchange agent 12. Giese teaches that the role of the exchange agent 12 is "to establish and manage end-to-end transport of payload data between the involved parties" (col. 12, lines 33-35). That is, the exchange agent 12 does not produce any payload data. Instead, the exchange agent 12 merely selects for each service instance "a best match set of transport services for the service instance" (col. 3, lines 63-64). According to Giese, the exchange agent 12 includes (col. 4, lines 50-57):

> a service management portion adapted to select a respective best match end-point device for each party involved in the service instance; a directory services portion adapted to resolve a physical address on the network corresponding to each selected end-point device; and a session control portion adapted to select, from the network space, a best match set of transport services for end-to-end connectivity between the selected end-point devices.

None of the portions of the exchange agent 12 produces payload data, contrary to the Examiner's assertion. Accordingly, Giese does not teach or suggest that the exchange agent 12 performs a prescribed function to produce a service deliverable requested in the given request-for-service call, as recited in claim 1.

The Examiner also has asserted that the service triggers issued by the contact agent 10 correspond to the access file recited in claim 1 that invokes at least one service module (which the Examiner has identified as corresponding to the exchange agent 12). Contrary to the Examiner's assertion, however, the service triggers issued by the contact agent 10 do not invoke the exchange agent 12. Rather, Giese explains that:

> Based on the application session requirements (originating party goals) and profile information, the Contact Agent 10 can also issue triggers for value added communication services, such as, for example, security, virtual networking, mobility and brokering services.

That is, the service triggers only invoke value-added services; they do not invoke the exchange agent 12.

d. <u>Giese's transport agent</u>

The Examiner has identified Giese's transport agent 14 as corresponding to the communication module recited in claim 1. Claim 1, however, recites that the communication module receives a service deliverable in accordance with a hypertext transfer protocol. As explained above, however, the "Exchange agent 12 and adjacent networks 48 interact with the Transport Agent 14 using network primitives" (col. 16, lines 9-11). Giese explains that (col. 16, lines 28-46):

> Network primitives are similar abstractions that enable users, user agents, Exchange Agent 12 functionality or adjoining networks 48 to interact with the Transport Agent 14, and thus

> the Transport Services Layer 8 of the network 2. The Transport Agent 14, in turn, implements universal Class or Quality-of Service Primitives that permit network services to initiate, maintain and terminate communication paths over any of a variety of transport technologies. Network services are unaware of transport technology.

There is no hint in this disclosure that would have led one of ordinary skill in the art at the time of the invention to structure communications between the exchange agent 12 and the transport agent 14 in accordance with a hypertext transfer protocol. Indeed, as explained in the Overview above, the contact agent 10 and the exchange agent 12 are on a single machine and therefore do not communicate over a network connection. For this reason, there is no reason whatsoever for one of ordinary skill in the art to modify the communications between the exchange agent 12 and the transport agent 14 to be in accordance with a networking protocol, much less to modify such communications in accordance with an application layer networking protocol such as HTTP.

Claim 1 also recites that the communication module transmits the service deliverable to the destination network node specified in the given request-for-service call in any format selected from a voice format, an internet format, an e-mail format, and a wireless format. Contrary to the Examiner's assertion, however, Giese's transport agent 14 does not transmit service deliverables in accordance with an application layer format, such as a voice format, an internet format, an e-mail format, or a wireless format.

Giese teaches that the transport agent 14 (col. 4, line 64 - col. 5, line 6):

> [is] adapted to engage, from the transport services layer, a best match set of transport media to accomplish end-to-end transport of the payload data of the service instance, and comprising: a directory services portion adapted to determine a best-match route for end-to end connectivity across the communication network; and a session control portion adapted to engage, in the transport services layer, transport services layer devices corresponding to the best-match route.

Giese also teaches that (col. 14, line 57 – col.15, line 17):

> Like the Exchange Agent 12, the Transport Agent 14 provides network functions, and is divided into functional partitions providing the following services: Transformation Services 38, Service Management 40, Directory Services 42, Session (Communication) Control 44 and Brokerage 46 (See FIG. 8). However, in the Transport Agent 14, functionality is path

> associated. Path associated functions allow services to carry out end-to-end communication using path level abstractions.
>
> Exemplary path associated functionality in each partition is as follows: Transformation 38 includes path edge conditioning and proxy clients. Path edge conditioning consists of the control adaptation, message generation and message interpretation required for inter-working between external systems and the ECS layer 6. Proxy clients adapt non-conformant legacy external systems (such as PSTN POTS) so that they can utilize the capabilities of the ECS layer 6. Service Management 40 consists of service configuration, account and policy management. Network management is just another service, albeit a privileged one, that is substantially automated through contracts as will be described below in more detail. Directory Services 42 encompass route selection and interoperability resolution for equipment and services in the communication path. Session Control 44 provides path session activation mechanisms. Brokerage services 46 enable selection of end-to-end communication paths of the appropriate qualities-of-service and provides assembly of contract details for the enhanced sub-layer.

Nowhere in Giese's disclosure is there a teaching or suggestion that would have led one of ordinary skill in the art to modify Giese's transport agent 14 to transmit a service deliverable received from the exchange agent to the destination network node specified in the given request-for-service call in an application layer format selected from a voice format, an internet format, an e-mail format, and a wireless format. Indeed, the transport agent 14 only engages the set of transport services selected by the exchange agent 12 at the level of the network services layer 8. This process involves adaptations and transformations at the hardware (e.g., node and switch) level to control "the physical movement of the payload data" (col. 9, line 38). The actual application layer format of the payload data is not affected by the transport agent 14.

e. Conclusion

For at least the reasons explained above, the Examiner's rejection of independent claim 1 under 35 U.S.C. § 102(e) over Giese now should be withdrawn.

B. Claims 3-14, 16, and 19-20

Each of claims 3-14, 16, and 19-20 incorporates the features of independent claim 1 and therefore is patentable for at least the same reasons explained above.

C. Claims 21-23 and 25-27

Each of claims 21-23 and 25-27 incorporates the features of independent claim 1.

The Examiner has rejected claims 21-23 and 25-27 under 35 U.S.C. § 103(a) over Giese in view of Angwin (U.S. 6,477,576). Angwin, however, does not make-up for the failures of Giese's disclosure to teach or suggest the features of independent claim 1 discussed above. For at least this reason, the Examiner's rejection of claims 21-23 and 25-27 under 35 U.S.C. § 103(a) over Giese in view of Angwin should be withdrawn.

In addition, the Examiner has asserted that:

> Giese does not specifically teach or suggest the step of converting VoxML or WML format request-for-service call into a HTTP format request-for-service call and transmit it to access file. However, Angwin teaches the step of converting VoxML or format request-for-service call into a HTTP format request-for-service call and transmit it to access file [col. 7, lines 25-37; col. 8, lines 48-55]. It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Giese and Angwin because doing so would bring convenience to user by providing service to the user whose device has voice-only capability or the display screen is small [Angwin, col. 7, lines 31-33]. One of ordinary skill in the art would have been motivated to modify Giese's system with Angwin's converting step to attract more users.

The Examiner, however, has misconstrued Angwin's disclosure. In particular, Angwin does not teach or suggest anything about converting a request-for-service call in a voice or wireless format into a hypertext transfer protocol request-for-service call.

In the first section cited by the Examiner (col. 7, lines 25-37), Angwin merely teaches that the server 20 responds to a Request Services Menu message received from a requesting device by providing to the requesting device a menu of services tailored to the characteristics of the session with the requesting device. This section certainly does not teach that the service coverts the Request Services Menu message into an HTTP Request Service Menu message, as asserted by the Examiner. Indeed, such a conversion would not serve any useful purpose whatsoever since the server 20 has already received and interpreted the Request Services Menu message.

In the second section cited by the Examiner (col. 8, lines 48-55), Angwin merely teaches that the server 20 may provide the requesting device with a complete menu of services or a URL pointer to a services menu. Angwin also teaches that, in the case where the server merely provides a URL pointer, the requesting device may obtain the corresponding services menu by issuing, for example, an HTTP request for the specified URL. This section certainly does not teach that the service coverts the Request Services Menu message into an HTTP Request Service Menu message, as asserted by the Examiner. Indeed, such a conversion would not serve any useful purpose whatsoever since the server 20 has already received and interpreted the Request Services Menu message.

III. Conclusion

For the reasons explained above, all of the pending claims are now in condition for allowance and should be allowed.

Charge any excess fees or apply any credits to Deposit Account No. 08-2025.

Respectfully submitted,

Date: July 13, 2004

Edouard Garcia
Reg. No. 38,461
Telephone No.: (650) 631-6591

Please direct all correspondence to:

Hewlett-Packard Company
Intellectual Property Administration
Legal Department, M/S 35
P.O. Box 272400
Fort Collins, CO 80528-9599